에너지-지향 달빅 바이트코드 스케줄링 기술

고광만*, 박희완**, 윤종희***, 최광훈****
*상지대학교 컴퓨터정보공학부, **한라대학교 정보통신방송공학과
***강릉원주대학교 컴퓨터공학과, ****연세대학교 컴퓨터정보통신공학부
e-mail:{kwangman.ko, heewan.park, skyglory, kwanghoon.choi}@google.com

Energy-oriented Dalvik Bytecode Scheduling Technique

Kwang-Man Ko*, Hee-Wan Park**,
Jong-Hee Youn***, Kwang-Hoon Choi****
*School of Computer and Information Engineering, Sang-Ji University
Computer&Telecommunication Engineering Division, Yonsei University
***School of Info. & Comm., Broadcasting Engineering, Halla University
****Dept. of CS & Engineering, Gangneung-Wonju National University

요 약

안드로이드 플랫폼에 적합한 어플리케이션 보급이 급증하면서 안드로이드 가상머신인 달빅(dalvik)의 성능 향상을 위한 연구가 다양하게 시도되고 있다. 전력 공급이 제한적인 모바일 기기에서 효율적인 어플리케이션 실행을 위한 플랫폼의 성능 향상과 더불어 전력·에너지의 최적화된 소비가 중요한 이슈가 되고 있다. 이 논문은 달빅에서 실행되는 dex 파일의 바이트코드를 에너지 소비 중심으로 스케줄링하여 Java 어플리케이션의 전력·에너지 소비를 최적화하고자 하는 시도이다. 에너지 지향적인 스케줄링 기법은 전통적인 리스트-인스트럭션 스케줄링 기법을 기반으로 하였으며 스케줄링 전·후의 실험 결과를 제시하여 이 연구의 효과를 입증한다.

1. 서론

모바일 기기 사용의 증가는 플랫폼에 적합한 소프트웨어 어플리케이션을 빠른 시간에 개발하고자 하는 요구에 부응하여 운영체제, 미들웨어, 라이브러리 등과 같은 소프트웨어 개발 환경의 소스를 공개하는 정책이 확산되고 있다. 오픈 소스 정책은 소프트웨어 개발 및 관리의 주체가 공급자에서 소비자로 이동하는 페러다임으로 오픈 소프트웨어의 소스 코드는 모두가 이용 가능하다. 이러한 이유로, 소프트웨어 배포 능력을 향상시키고 발달시키기 위해 구글의 안드로이드 플랫폼과 같은 공통 플랫폼이 공개되고 있다[1].

안드로이드 어플리케이션은 안드로이드 가상머신인 달빅(dalvik) 인스턴스와 함께 자신의 프로세스에서 실행되며 달빅은 여러 인스턴스가 효율적으로 실행될 수 있도록 구현되어 있다. 달빅에서 Java 어플리케이션(*.java)을 실행하기 위해서는 Java 컴파일러에 의해 생성된 클래스 파일(*.class)을 최소화된 메모리 영역을 차지하는 최적화된 Dalvik Executable(*.dex) 형식의 파일로 변환해야 한다. 달빅은 레지스터 기반의 가상머신으로서 dx 툴을 이용하여 Java 클래스 파일을 dex 형식으로 변환한다[2].

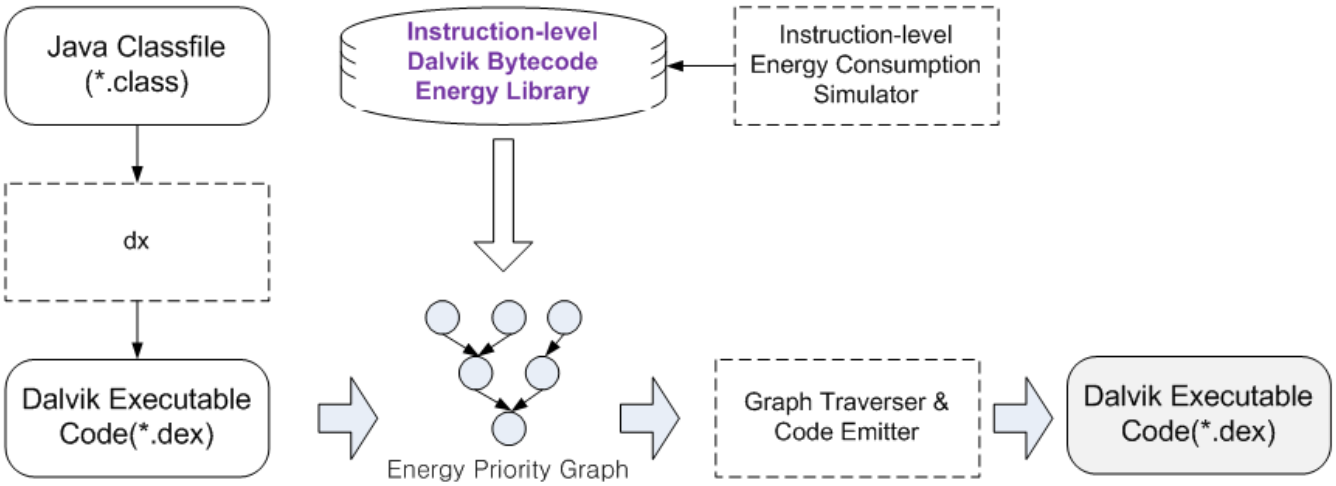
최근까지 에너지-지향적인 컴파일러 최적화 동작을 통해 어플리케이션의 에너지 소비를 줄이기 위한 노력이 매우 효과적으로 진행되어 왔다. 고성능 컴퓨팅 환경은 빠른 실행 속도, 적은 메모리 사용량 등에 집중되어 평가되었지만 최근에는 저 전력·에너지 사용과 이를 위한 다양한 소프트웨어 관리 기술이 강조되고 있다[3].

이 논문에서는 dex 파일에서 바이트코드를 추출한 후 전통적인 리스트-인스트럭션 스케줄링 기법을 기반으로 에너지 소비를 최적화하기 위한 바이트코드 스케줄링 기법을 제안하고 이를 검증한다. 이 논문의 구성은 제 2장 본론에서 연구 접근 방법과 내용에 대해 기술하고 최종적인 실험 및 분석 결과는 현장 발표를 통해 소개한다. 제 3장에서 이 논문의 의미와 향후 연구 내용에 대해 기술한다.

2. 본론

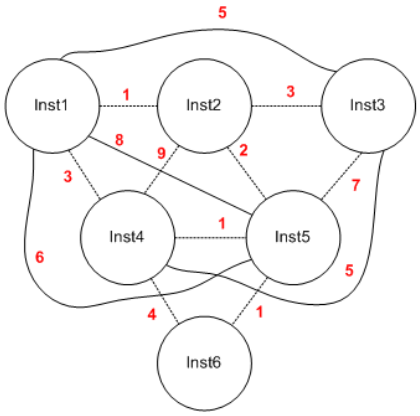
소프트웨어 수준에서 어플리케이션이 소비하는 전력을 정확히 측정하고 분석할 수 있는 접근 방법은 아키텍처의 각 컴포넌트와 수행 기능을 모델링한 후 전력 소모 모델을 추가하는 시뮬레이션 방법으로서 저수준 시뮬레이터와 고수준 시뮬레이터로 구분된다[4]. 고수준인 인스트럭션-수준(instruction-level) 시뮬레이터는 하나의 인스트럭션이 실행될 때 소비되는 에너지 소비량(base cost)과 하나 이상의 인스트럭션이 실행될 때 소비되는 에너지 소비량(inter-instruction cost)을 반복적인 시뮬레이션을 통해 측정된 전류(I)와 전압(Vcc)을 이용하여 전력(P=I×Vcc) 소비량을 계산한다. 이를 기반으로, 어플리케이션 실행시간(T)에 의해 소비되는 에너지 소비량(E=I×Vcc×N×τ)이 계산되며 실행시간(T=N×τ)은 클럭수(N)와 클럭주기(τ)를 이용하여 결정한다[5].

전체 시스템은 [그림 1]과 같이 구성되며 달빅 실행파일의 바이트코드에 대한 에너지-지향 스케줄링을 통해 에너



[그림 1] 전체 시스템 모델

지 소비 최적화된 달빅 실행파일을 생성한다. 이를 위해, 우선적으로 달빅 바이트코드 스트림을 추출한 후 기본블록 단위로 제어 및 자료 흐름 그래프를 구성한다. 바이트코드간 에너지 소비 정보를 가중치로 갖는 에너지-가중치 연결 그래프(Energy-weighted Connected Graph)를 [그림 2]와 같이 구성하기 위해 [6] 연구를 인용하여 바이트코드 대한 에너지 소비 정보를 라이브러리 정보 구성에 활용한다.

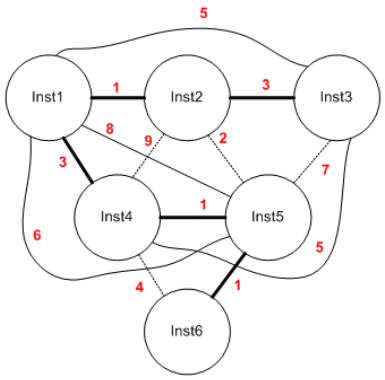


[그림 2] 에너지-가중치 연결 그래프

최종적으로 최적화된 에너지 소비값을 갖는 스케줄링을 위해 [그림 3]과 같이 에너지-지향 우선순위 그래프를 구성한 후 이를 방문하면 코드를 생성한다.

3. 본론 및 향후 연구

이 논문은 안드로이드 달빅에서 실행되는 dex 파일의 에너지 소비를 최적화하기 위한 시도로서, 달빅 바이트코드간 에너지 소비량을 가중치로 갖는 에너지-가중치 그래프로부터 스케줄링된 코드를 생성한다. 현재, 이 연구의 정확성과 활용성을 높이기 위해 달빅 바이트코드의 에너지 소비량 측정 시뮬레이션과 에너지-가중치 그래프 구성에 대한 보완연구가 진행 중이며 최종적으로 에너지 소비량 감소된 실험 결과를 제시할 예정이다.



[그림 3] 에너지-지향 우선순위 그래프

참고문헌

[1] Software Engineering Research Group, “Analysis of Dalvik Virtual Machine and Class Path Library”, Inst. of Management Science Peshawar, Pakistan, 2009.
[2] Dalvik Virtual Machine. http://www.dalvikvm.com/.
[3] A. Parikh, Soontae Kim, M. Kandemir, N. Vijaykrishnan, M. J. Irwin, "Instruction Scheduling for Low Power", Journal of VLSI Signal Processing, Vol 37, pages 129~149, 2004.
[4] I. Kadayif, M. Kandemir, G. Chen, N. Vijaykrishnan, M. J. Irwin, A. Sivasubramaniam, “Compiler-directed high-level energy estimation and optimization”, ACM Transactions on Embedded Computing System, Vol. 4, Issue 4, 2005.
[5] Vivek Tiwari, Shard Malik, Andrew Wolfe, Mike Tien-Chien Lee, “Instruction Level Power Analysis and Optimization of Software”, Journal of Signal Processing, Vol. 1, Num. 18, 1996.
[6] Kyuwon Choi, Abhijit Chatterjee, “Efficient Instruction-Level Optimization Methodology for Low Power Embedded Systems”, ISSS 2001, 2001.